

Still Alive With Sir Clive!

# ZXir QLive Alive!

The Timex/Sinclair North American User Groups Newsletter

Volume 10 No. 2

Summer 2000

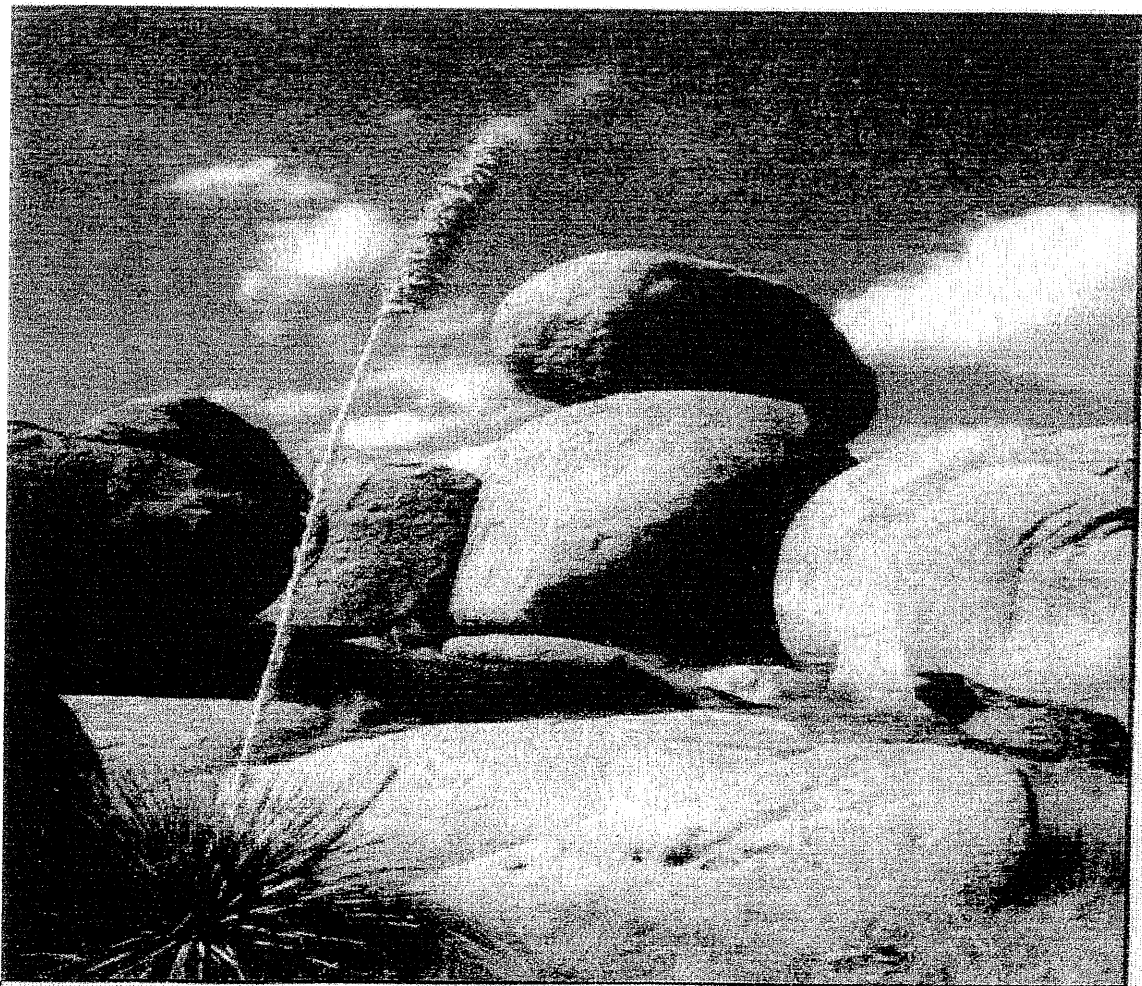
## MEMORY MAP

### ADDRESS

- |   | <b>Routines</b>                               |
|---|---|
| 2 | Information and Chairmen — Treasury Notes     |
| 3 | <i>Input/Output</i> — by Abed Kahale          |
| 6 | CATUG   |
| 6 | Sinclair E-Mail List                          |
| 7 | QL Users E-Mail Database                      |
| 8 | From the Chairman's Disk — Donald Lambert     |
|   | <b>FILES</b>                                  |
| 9 | 7-bit ASCII Text Filter TS-2068 — David Solly |

### ADDRESS

- |    | <b>FILES</b>                     |
|----|----------------------------------|
| 11 | How to Hack on the ZX Spectrum   |
| 14 | Memory Improves With Age — WDJUP |
| 15 | Bank Switching — WDJUP           |
|    | <b>SUBROUTINES</b>               |
| 18 | Unclassified Ads                 |



ZXir QLive Alive! ©

Established 1991 The Timex/Sinclair North American User Groups Newsletter

# T/SNUG Information

We wish to support the following platforms : ZX-80/81, TS-1000, Spectrum, TS-2068, Z88 and QL. If you have any questions about any of these fine Sinclairs, contact the:

## Chairman

Chief Motivator  
Donald S. Lambert  
738 Gunnar Ln.  
Forsyth, IL 62535  
(217) 875-8043

## Vice-Chairmen

### Z88 Library

Dave Bennett (HATSUG)  
1275 Timber View Dr.  
Mechanicsburg, PA 17055-9146  
717 732-4374

### QL Hacker's Journal

Timothy Swenson  
2455 Medallion Dr.  
Union City, CA 94587-1914  
swensontc@geocities.com

### TS-2068

Rod Humphreys (VSUG)  
10984 Collins Pl.  
Delta, BC V4C 7E6 Canada  
604 583-2819

### QL PD Library

John Donaldson (CATUG)  
835 Foxwood Cir.  
Geneva, IL 60134-1631  
630 232-6147

### AERCO & Z80 Emulator

Keith Watson  
41634 Amberly Dr.  
Mt. Clemens, MI 48038

### BBS ---GATOR---

Bob Swoger (CATUG)  
613 Parkside Cir.  
Streamwood, IL 60107-1647  
630 837-7957 Work 847 576-8068

Any of the above can also be reached by E-Mail through the **Club BBS 847 632-5558**

## ZXir QLive Alive!

Is the newsletter of T/SNUG, the Timex/Sinclair North American User Groups, providing news and software support to the T/S community in a VOLUME of four newsletters per year, beginning with the Spring (March) issue.

T/SNUG's main goal is to preserve and encourage the use of Sinclair computers by providing an open forum for the exchange of knowledge, building and maintaining of software libraries. Providing vendors, repair service and members with free ad space.

It is the user groups and individual subscribers, rather than the vendors, that provide the pecuniary support for this newsletter. Vendors and developers receive this newsletter free of charge, though contribution from vendors and user groups is gratefully accepted. Please support our vendors and service providers whenever possible.

If you have a problem or you have solved a problem, please share it with the rest of us. No problem will be considered unimportant.

## Editor/Treasurer Publisher

You can keep T/SNUG alive by an annual contribution of \$12 for one VOLUME made payable to Abed Kahale. Send check to:-

ABED KAHALE  
3343 S FLAT ROCK CT  
SIERRA VISTA AZ 85650-6874  
520 378-3424

Back copies are available for \$1.00 each postpaid.

## Trea\$ury Note\$

As of June 8, 2000, we have a balance of \$903

## Article Contributions

Send in your articles by disk, hardcopy mail, or e-mail and your inputs to:—

Abed Kahale

E-mail: AKahale@compuserve.com

## Welcome

## CATUG Members

## GATOR'S

## Twisted Pair

To better inform the Sinclair Community, three 24-hour a day BBSs are now provided to serve you. You are encouraged to exchange mail and use the files sections of these boards. Bulletins and ads are available to all.

**Q-Box BBS 810 254-9878**  
Utica, Michigan

**SOL BBS 520 882-0388**  
Tucson, Arizona

**Club BBS 847 632-5558**  
Arlington Heights, Illinois

## WEBPAGES

<http://users.aol.clubbbs/tsnug/>  
<http://www.outlawnet.com/~jboatno4>

If you know the Internet E-Mail address of a Sinclair user, but do not have access to Internet, simply address your E-Mail to GATOR Sinclair on the 24-hour Club BBS and include the name and E-Mail address of the user you wish to reach. Then check the Club BBS from time to time if you expect a reply.

We encourage you to exchange mail and contribute to the UPLOAD section. Call and register using your first, last name and phone number along with a password you won't forget. **Write It Down!** Do not try to do anything else at this time.

When you call-in the next time, you will have Level 5 security and be able to enjoy full user privileges. The BBS has smaller sections called conferences. Select "J" for "Join a Conference". Select "TIMEX" to get into the Sinclair Section. The mail you then read will only be from other T/S users. Use extension ART for articles, ADS for ads and NWS for news when UPLOADing.

For help, contact the SYSOP, Bob Swoger, by leaving a message, mail, E-Mail or phone.  
CENG108@email.mot.com

# Input/Output

*by Abed Kahale*

Hello Abed

That last jpeg (diag. 4) doesn't have the minor correction of the missing "INT" after the "<>". Sorry for causing you the trouble of a correction. Take care,

**Joe Rampolla**

What a wonderful website! I've been T/S'n since 1984. My system includes a 2068 LarKen disk system 3.5 and 5.25 floppy, Aerco Parallel, Westridge (silver case) 2050 modem, Haven't gotten around to building the RGB interface yet(10yrs...)I'd like to build up a 1000. **I have a box full of them with printers and 4 or 5 modem boards**, but I need at least one RS232 port made so I can do 1200baud and upload software to P.C. more later

**Neil**

(904)740-9101 Printmaxx pxx1@netzero.net

Hello Abed:

In reply to your kind letter of Nov. 21, I have just gotten off the dime and sent off seven boxes of Sinclair stuff to Jack Boatwright in Bend, Oregon, including ZX-81's and a Timex 1000 with extra memory, a TS2068 with 5-1/4 floppies using Aerco system plus lots of old publications and spec. sheets. I talked with him before sending this load and he really seemed to want it. He is only 53 years old so he should have a good chance to put together a history of Sinclair computers. He started with a ZX-81 and then later a TS-2068 Just like me and he also has a QL but had not used it yet. He said that he would share the things he did not need with others who wanted them.

I have not any further information about getting a working copy of Assembler Work Bench by Zita Soft for my QL. I have the program copied to my floppy but it needs the original tape as the security key. The original tape jammed so it was useless and I cannot load the program without the security key. This is a QL program that came on micro-tape. I wrote a letter to Zita Soft in England but they never replied. They must be defunct. Is there someone I should write to in Europe that might be helpful or the internet?

Yes, I have to get up to date and get on the internet so we all can communicate with each other. I have the equipment with Windows 95 but I have to make space to assemble it with printer, etc. That is why I have been cleaning out old computer stuff to make room for new computer stuff that will quickly get out of date. I am keeping my QL which works well and I can still program it easily. Are you keeping the ZXir QLive Alive still?

**John Pegram**

1126 Stradella Rd.  
Los Angeles, CA 90077

I contacted Bill Cable, and he may have a copy of Assembler Workbench, but he suspects his tape is in the same condition.

**Al Feng**

RE: **Byte-Back** Advertisement

I didn't realize you were still running the ad. I'm still

here and still have an attic full of stuff, but haven't heard from anyone in two years from the ad. Don't bother wasting the space. Thanks

**David Leech**

Hey! I just visited your Site

<http://www.outlawnet.com/~jboatno4/welcome.htm>, and found it quite interesting. I'm running this program called **Hamster ADS** which pays up to 20 cents per unique visitor & 5 cents per click on TEXT ADS which can fit right into your site perfectly.

We cut checks to our webmasters EVERY Friday and I'm sure you will be very interested.

Check us out @ <http://www.hamster.com>

If you have any questions email me at [jack@hamster.com](mailto:jack@hamster.com)  
Hope to see you on our program soon!

**Jack**

<http://www.hamster.com> [jack@hamster.com](mailto:jack@hamster.com)

Hello Abed:

Thank you for your letter of March 23, 2000 in which you contacted John Donaldson and Al Feng. I hope one of them may have a working micro-tape for the Zita Assembler Workbench. I will write to each of them to see what they may have to say.

As you know I am still using my QL and will still use it even after getting my used PC up and running. So I will be interested in all that goes on in the Sinclair world. With best wishes,

**John B. Pegman**

Abed;

I hope this finds you and yours well. I forgot to include my email address in the article on the interface. Have you heard anything from the TS community that shows an interest in experimenting with this puppy? I need to know if I should press on with a PCB or tinker on with my proto.

I have been rather busy lately and will be until the end of April (my son is getting married then!) so I won't have much time to work on the 2068 stuff. I have been proceeding with my disassembly of the 2068 ROMs to see how to load and save using other "channels." I have sussed most of the EXROM routines that do this. I have also found the routines for the ERASE, FORMAT, CAT, and MOVE keywords - they just call the ON ERROR handler if interpreting. I have been toying with the idea of modifying the present 2068 firmware to make it better use the interface. I am thinking that the expanded 2068 would use an OS that was optimized for a bank-switched machine and have better hooks to add device drivers.

I have been playing with my Z88 lately and have been looking at its OZ operating system and thinking that there is much there that is useful for the expanded 2068 - especially considering that the Z88 was designed from the ground up as a bank-switched machine. In addition, it would be interesting to run Pipedream or other well behaved Z88 applications (those that use only OZ calls) on the 2068.

**Jeff Burrell**

[JBurrell@endocardial.com](mailto:JBurrell@endocardial.com)

Hi Abed,

Just a quick note to let you know my new email address a bit shorter and easier to remember.

I hope all is going well for you and yours I am doing fine. I still get quite a few calls re: TS items. I refer most of them to Jack Boatwright or to you. I just send a fellow named Robert Webster from St. Louis to both of you. He wants to get back into TSing.

Thanks again for keeping me on the ZQA! mailing list via email. I enjoy reading (i.e. hearing) the items. I really miss not being able to use a TS computer. Later,

**Rod Gowen**

aw723@osfn.org

Hello Abed, I've gotten my Spectrum computer back from John R. Rish. I still have one little problem, I need an eight pin monitor cable for a composite monitor. Got any idea's where I can find one, or should I run an ad, in the next ZXir QLive Newsletter. It would be great to see it run on a monitor. Thank You!

**Dane Stegman**

26 MARSHALL AVE  
AKRON NY 14001-1016

PS: It does seem to power up okay, but I need a monitor cable to get it up and running right!!!

danesteg@juno.com

Mr. Abed, I got your e-mail address from Rod Gowen, he said you could put me in touch with John Shepard, I need a TS-2068 computer mine has gone out on me. I get a screen that comes up to the Timex copyright screen and goes no further, if you would please be kind enough to help out I would be very thankful. I talk to Dan Elliott and he said that it was most likely the big square chip on the board which he also says they are no longer aviable, so if you could get me another TS-2068 I would be very happy...

**Robert Webster**

7831 Weaver Ave.

Maplewood, MO 63143

(314) 645-5267

"rwebs@freewwwweb.com

"@freewwwweb.com

Here are the two addresses you should try:

John (Jay) Shepard

<jshepard@netins.net >

Jack Boatwright

<jboatno4@outlawnet.com >

I can also have your request in the next

(June) Newsletter, if you don't get one let me know.

Dear Abed,

I have a completely new TS2068 system still in the box, computer, TS2050 modem, TS2020 recorder, printer

and all the software. It has been sitting on my closet shelf for years. Do you know if any of your contacts need such a system. Sincerely,

**Tom Parks**

tparks@alhloghomes.com

www.alhloghomes.com/loghomecare

Appalachian Log Care

11312 Station West Drive

PH: 1-800-726-0708

Knoxville, TN 37922

FAX: 423-675-2662

Thanks for sending me the info. I have already sent Mr. Parks e-mail, and called the 800 number and left a message, thanks again.

**Robert Webster**

To: dkl@dpliv.com

Good to hear from you, Dave!

No, I have never heard of the 2040 used with anything other than the Sinclair computers.

Have you seen our Timex Homepages linked from:

<http://members.aol.com/clubbbs>

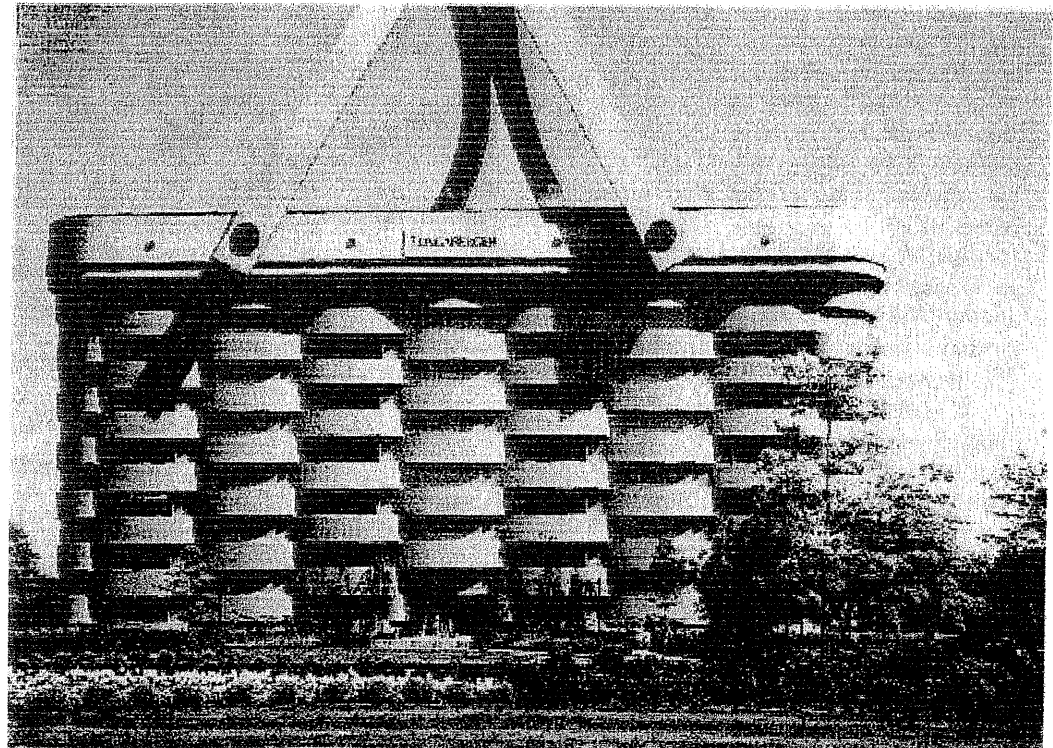
Dave, it just came to me, there is an Alfacom thermal printer that uses 8" thermal printer that uses generic printer commands from an IBM platform machine. Does that interest you? There may also have been a 4" version.

Dave Lebowitz, found his old TS2040 printer and wondered if anyone had ever hooked a TS2040 to an IBM platform

==GATOR==

Dear Abed,

You continue to do a most excellent Job in getting the



Longaberger Basket Co. Near Newark, Ohio  
World largest basket Guinness Book etc...

Newsletter out with so much professionalism. Your command of the American Language is superb and all of the articles in the newsletter are interesting. I appreciate the efforts of each and every contributor and our Chief Motivator. Please keep up the good work! Best wishes,



**Hugh Scriven**

*Thank you, Hugh.*

Dear Abed,

Lately I've become interested in USB transmission of data between PC's and I recalled that in an early issue of "ZXir QLive Alive" you had written an article that described a set-up that allowed a TS2068 to talk to a PC. After looking through numerous earlier issues of your newsletters ... I found it (in Vol 3 #4 -- Winter 1993).

While searching through the issues I couldn't help but wonder if anyone had considered cataloging the articles in your newsletter? AND has anyone designed a practical computer control for an electronic organ?

Most of them operate on low voltage (transistorized or digitized) so it shouldn't be that difficult and it would seem to me that it would result in saving a lot of Timex computers from the scrap pile! OK ... "LECTURING MODE OFF"!!! Sinclairly,

I'd be willing to collaborate with anyone interested in writing an article. My main knowledge of computers, which is quite limited was with the T/S 2068, and my "hands-on" knowledge of electronic organs was with the Schober and Devtronix organs ... but I have access to info on a few others. Sincerely,

**Fred Henn**

FRED J HENN

230 N FRENCH RD

BUFFALO NY 14228-2033

oranur@juno.com

Thanks for the info, Abed ... I do appreciate it. Tiger Computers is touting it as, The World's Hottest Operating System AND the wave of the future! One of my reasons for being interested is that "Uncle" Clive's new laptop, will be marketed with Linux? Sinclairly,

**Fred Henn**

*> Linux is excellent for those who like to get into a different OS than Windows and do have sufficient computer knowledge to do the work with it. The free OS (by Linus Torvalds) is not a "plug-n-play" and programs made for Win will not run in it. But the commercial version that I have seen advertised makes life easier for about \$50.*

US QL 2000 Show

Sponsored by QUANTA and NESQLUG

White River Junction, Vermont USA on May 20, 2000

Bill (Cable)

I put portions of your e-mail on the T/SNUG Homepage, Bill.

You picked the same day as the Glenside Color Computer Fest that I have to work on! I shall update the T/SNUG homepage to make your ad even nicer as when I have time-soon!

Thanks for the input and I hope you have great success - wish I could be there. This years QL show sounds better than last year's.

**Robert Swoger**

<http://members.aol.com/clubbbs/tsnug> ---GATOR---

I wrote to Pegram last week ...

I contacted Bill Cable, and he may have a copy of Assembler Workbench, but he suspects his tape is in the same condition.

I received an e-mail from Bill Cable Monday morning and he said that he would be able to send Pegram the requested tape ...

**Al Feng**

*Thank you, Al*

Thanks Bob,

That is very nice. You put some work into that (clubbbs@aol.com). Should help get a few more attendees. It looks like we will have a good show. Lots of QL notables will be there. Be nice to see you there too. Does your Gator nickname have anything to do with University of Florida. I graduated from there in 1967.

**Bill Cable**

The nickname GATOR was given to me by a member of the Motorola Computer club when I joined in 1984 and put a BBS on their PDP 11/60 Computer. They said "I made the swamp livable".

---GATOR---

Hi Bob,

I don't know if anyone is going or not, but I sent a copy of your e-mail to John. Dennis might be interested but I don't know what his e-mail address is right now. I will send him a copy as well when I find out what it is. We don't have the meetings around here like we use to. Too bad, I liked them.

**Keith Watson**

Thanks for the feedback, Keith. I sure do miss those get-togethers, also. We have not had a meeting down here since last September. I also miss them. There is still TS2068 stuff I'd like to do but the TIME BANDITS keep me from doing these projects. Did you visit the T/SNUG homepage yet?

---GATOR---

Hello Abed.

Last week, I received an original and a copy of Assembler Workbench from Bill Cable. The program works perfectly! Thank you so much for finding someone who could supply this program to me. I am writing to Bill to thank him and send along some reimbursement for the program.

I am still slowly cleaning out my upstairs room and have sent all my ZX81, TS-1000 and TS-2068 stuff to Jack Boatwright up in Oregon. However, I am keeping my QL with all its programs as I can program it easily. I am making room for a PC with Pentium P-90 complete with Windows 95. Then I can get on the internet and communicate with others who have such an address. With best wishes,

**John Pegram**

Hi Abed,

It was mandatory for me to change from @compaq.net to @email.msn.com since @compaq.net will be discontinued as of the 13th May. On the 8th the new CD arrived (I pay nothing till my year is up) for the change over but the CD would not work so I finally got on line with it after I made several 800# calls. And from Monday 8th for two days I could not access any email. Then I was referred to [www.mailstart.com](http://www.mailstart.com) where I could get my @compaq.net mail. And for another two days I could not send email till I found out how to do it through mailstart. I finally got a nice lady at Compaq to step me through the procedure to have

both @compaq.net and @email.msn.com email picked up automatically. I think that is working.

The time I planned to spend writing my column was spent trouble shooting the changeover. I will get onto it as soon as I can what with honeydo projects having a high priority here. I guess ol' Murphy is laughing at me.

So don't forget to put my e-mail address in ZQA as dslambert@email.msn.com.

I was perfectly happy with @compaq.net but it seems that @email.msn.com is faster

**Don S. Lambert**

dslambert@email.msn.com

Sooooo, I finally got off my lazzzy buit and sent some folks some stuff.

Luke Perry got Joan Kealy's modem and a grab bag of tape SW, with emphasis on DTP. ...wanted to send him Lemke's stuff but couldn't find a full suite w/o pawing through more than two boxes, but will pursue that.

Al Feng got a QL to replace one that bit the dust.

And ...Gil Parrish got the only AJ Microdrive w/ a bunch of stringy floppies ... many with SW.

I sent all by USPS, with tracking numbers, so via the Web so I knew when they received their goodies.

I've heard from the first two, but not from Gil. Will ask how it's working after sending this.

Hope all is well with you and yours and this crazy end-of-the-world weather will settle down for what looks like a very hot summer.

**Jay Shepard**

jshepard@netins.net

Guys, I am sooo glad to hear that the store is finally working out for the Sinclair community needy. Jack, I wonder if you have similar items to report? This would be good input for the newsletter. Also guys, is there a spare 16K Sinclair RAM pack in one of the stores? This request comes from duncan@military.com (Duncan Haberly)

--- (GATOR) ---

Hello!

Sorry my bad English I'm from Argentina, and I have a TK85, clone from the TS1500 I have tapes for TK, and convert this to .p files (for the Xtenders, ZX81 emulator) the problem is the

## Fortress of Zorlac

I really love this game, but the tape is corrupted, look for it in internet, but I could not find it. Would you know where to find it? I will thank him very much a help. Greetings from Argentina

**Juan Carpio**

juanchuscar@yahoo.com

Contact Antonio Castro castrox@portoweb.com.br

Thanks abed, really thanks you if you need something, send me an e-mail, please

juan

## CATUG

### Now DIRECT ZQA! MEMBERS

**John Donaldson, QL**

835 Foxwood Cir

Geneva IL 60134

ildnldsn@aol.com

6302326147

**Mark Dorinson, TS2068**

PO Box 351

Frankfort IL 60423

74200.257@compuserve.com

7087551833

**Cy Herre, QL**

666 S Bluff St 802

S Beloit IL 61080-2166

cyherre@aol.com

**Phillip Kwitkowski, TS2068**

2106 Dover Ln

St Charles IL 60174

kwit80@aol.com

6305846710

**Dean Mikolajczyk, TS1000**

4714 Arbor Dr #207

Rolling Meadows IL 60008

deanm97493@aol.com

**Frank Mills, TS2068**

417 S 47th Ave

Bellwood IL 60104-1723

effem@yahoo.com

**Larry Sauter, TS2068**

7747 W Bryn Mawr Ave

Chicago IL 60631

7737635383

**Lee Thoresen, TS2068**

45 Hickory Ln

Crete IL 60417

74200.257@compuserve.com

7086729066

--- (GATOR) ---

**Robert Swoger, LarKen, TS2068**

613 Parkside Cir

Streamwood IL 60107-1647

clubbbs@aol.com

6308377957

## Sinclair E-Mail List

Anderson, Paul	pandersn@peakpeak.com
Anson, Gerald	jerrya@aztec.asu.edu
Barker Robin	robin@di-ren.demon.co.uk
Bennett, Dave	Dbennett7@freewwwweb.com
Boatwright, Jack	jboatno4@outlawnet.com
Boehm, Al	boehm@ziplink.net
Boehm, Bill	boehm@plh.af.mil
Burrell, Jeff	JBurrell@endocardial.com
Burt, Richard	ajb@intranet.ca
C. A. T. S.	mf0002@epfl2.epflbalto.org
Cable, Bill	cable@cyberportal.net
Carpio, Juan	juanchuscar@yahoo.com
Castro Antonio	castrox@portoweb.com.br
Catotti, Christopher	kd4ace@compuserve.com
Chambers, George	gfchamb@pathcom.com

Collins, Bill	bcollins@home.ifx.net
Cottrell, Les	jacottre@gte.net
Cruz-Figueroa, Jaime	cruzfiguer@aol.com
Dansby, Andrew	adansby@atlantic.net
Davis, Frank	fdavis@iquest.net
Donaldson, John	ildnldsn@aol.com
Dorinson, Mark	74200.257@compuserve.com
Dunbar, Douglas	dldunbar@prodigy.net
Dunnet, Ron	ron@qubbesoft.freemove.co.uk
DuPuy, James	dupuy@pipeline.com
England, William	wengland@iname.com
Feng, Al	alfeng@juno.com
Fink, Mike	domino.cubes@excelsior.net
Fink, Mike	domino.cubes@pointblank.com
Firshman, Tony	tony@firshman.demon.co.uk
Franke, John	j.m.franke@larc.nasa.gov
Ganger, Gary	gangerg@dma.org
Gilbert, Robert	weena@netzero.net
Gillespie, Doug	aa431@cleveland.freenet.edu
Girnius, William	girnius_w@bls.gov
Gowen, Rod	aw723@osfn.org
Haberly, Duncan	duncan@military.com
Harbit, Ken	krh03@cvip.fresno.com
Henderlight, Mike	mikehend@microsoft.com
Henn, Fred	oranur@juno.com
Herre, Cy	cyherre@aol.com
Holmgren, Paul	paulholm@indy.net
Horton, Will	willhort@aol.com
Hoshor, Dave	dnhoshor@raex.com
Humphreys, Rod	rodh@lightspeed.bc.ca
Hunkins, James	jdunki@ibm.net
Impellizzeri, John	jimpellizzeri@compuserve.com
Jaap, Matthias	matthias_Jaap@hhs.hh.schule.de
Jonas, Mike	mjonas@bbn.com
Jones, Dilwyn	dilwyn.jones@dj.softnet.co.uk
Jones, Terry	tjones@iname.com
Kaczor, Jon	jazkaczor@aol.com
Kahale, Abed	akahale@compuserve.com
Kealy, Harriet Joan	hjkealy@admin.hilconet.com
Kenny, Larry	larken@storm.ca
Kingsley, Ed	edk4@aol.com
Knyszek, Theodore	thirteenth@worldnet.att.net
Kondrac, Mike	mkandrac@aol.com
König, Urs	urs.koenig@agrodata.ch
KurtK7	kurtk7@aol.com
Kwitkowski, Phillip	kwit80@aol.com
Lambert, Donald	dslambert@compaq.net
Lancaster, Garry	dharkhig@delphi.com
Lanciault, Francois	francois.lanciault@energies.alstom.ca
Lassov, David	emanon@azstarnet.com
LaVerne, Melvin	mlaverne@usit.net
Lebowitz, Dave	dki@dpdiv.com
Lessenberry, Gary	gl743@aol.com
Lewis, Jim	jlewis2@neo.lrun.com
Liebert-Adelt, Peter	p.liebert@t-online.de
Malloy, Bob	74776.2342@compuserve.com
McBrine, William	wmcbaine@clark.net
McKelvey, William	mckelveyw@delphi.com

Merz, Jochen	jmerz@t-online.de
Mikolajczyk, Dean	deanm97493@aol.com
Miller, Seymour	seymil@delphi.com
Mills, Frank	effem@yahoo.com
Muth, Bob	bobkeeper1@aol.com
Norton, Gary	gnorton@world.std.com
Parrish, Gil	gil.parrish@abanet.org
Pashtoon, Nazir	nazir.pashtoon@ingram.micro.com
Payne, Josh	joshpayne@bigfoot.com
Pazmino, John	john.pazmino@moondog.com
Perry, Luke	doidy34@yahoo.com
Perry, Russ Jr	slapdash@enteract.com
Radloff, Toby	tobyradloff@webtv.net
Rampolla, Joe	jprampolla@blazenet.net
Rigter, Wilf	rigter@cafe.net
Rish, John	74601.1535@compuserve.com
Shepard, Jay	jshepard@netins.net
Simon, Thomas	73177.333@compuserve.com
Skapinski, Thomas	tskapins@juno.com
Solly, David	ac355@ncf.ca
Stegman, Dan	danesteg@juno.com
Swenson, Tim	swensontc@geocities.com
Swenson, Tim	swensont@sirclive.csd.sgi.com
Swentko, Wally	wswentko@maroon.tc.umn.edu
Swoger, Robert	ceng108@email.mot.com
Taylor, Jeff	jetaylor@mdrobotics.ca
Thoresen, Jeff	74200.257@compuserve.com
Waldman, Stephen	brogine@hotmail.com
Walterman, Don	walterm@ix.netcom.com
Watson, Keith	keith_watson@juno.com
Wood, Roy	qbranch@qbranch.demon.co.uk
Zimmerman, George	gzimmer928@aol.com

## Users

### Email Database

Welcome to the Web Based  
QL Email Users Database.

Your email details have now been added to  
the database.

If you have a password you are now able to  
access the database and modify/delete your  
own record.

Should you forget/mislay your password, use  
the **\*forgot\*** facility on the database search  
page to email you a new one.

If you do not have a password, go to  
[www.quanta.uni.cc](http://www.quanta.uni.cc) and enter the database.  
Enter your email address and press submit.

You will then be emailed a password.

Regards

**Quanta Support**

**[www.quanta.uni.cc](http://www.quanta.uni.cc)**

Sender: [wwrun@www.jakinternet.co.uk](mailto:wwrun@www.jakinternet.co.uk)

# From The Chairmans Disk

Donald S. Lambert

**M**y plans to get things done T/S computer wise got sidetracked by honey do projects one of which was a biggie.

That was to do some landscaping in front of the house. If we had gotten the proper amount of rain the ground would have been soft. But it was hard as dry clay can be. I finally got that done although there were several trips to LOWE'S (which is about a half mile away) to get plants, border stones and such. A place nearby delivered a yard and a third of small river polished stones to put down, Not to forget all the trips to school events with our youngest grandchild and her activities in band, chorus and ballet. We will go to her graduation from middle school (8th grade) the twenty fifth of May.

**W**hile I did not get a copy of the program to accept the upload from the LASER PC3 to a PC I did get PC TOOLS 7.1 which includes such a program.

The set includes eight disks (720K 3.5) and ten manuals. When I have time that I won't be interrupted for several hours I will load them into the PC and try that out.

One of the interruptions has been tornado warnings wince we live in tornado alley. None have touched down and did damage near us one of the employees of STAPLES which is on this side of LOWE's saw a funnel cloud came down but it went back up before it got to our area. When we have storms I do not use the computer.

I have learned that most of my troubles with the PC have been the fact that they don't use the same words as Timex/Sinclair did. For instance a directory on T/S is a FOLDER on the PC and FAT (File Allocation Table) are disk sectors. There are other such changes of terms.

I lost a week when Compaq.net quit and I was switched to MSN. I had paid Compaq.net a year in advance and there is no money difference till after that is up. BUT, The CD that was sent to make the change **didn't** work.

I called Compaq tech and they didn't know what was going on. So I called MSN and I was requested to use the MSN signup that was on my computer in Windows Desktop. That got me loaded with MSN. However! I now had a password problem and a conflict of opinion by both Compaq.net and MSN as to who was in charge of getting me on line. And part of the trouble was the fact that I had used a 6 character password with Compaq not and I had to have at least an 8 character password with MSN. So I had added four characters to the

Compaq password and neither would accept the others password and they could not let me give a different one. For two days I couldn't access my e-mail and then the next two days I couldn't send any. Then I got a nice lady at Compaq to step me through the process to get that corrected. So far I have not had a problem with MSN.

I have checked out the Z88 and the DISK-88 (X0B) which is the software/hardware for the battery operated Radio Shack Portable Disk Drive 2 (battery operated) and found out that one of the two disk drives that I have has failed. But otherwise the interface and the disk drive operates correctly. But is it of a single sided 3.5 disk drive. A disk holds 202.24 K in two directory banks and of course it is a low density disk drive. Of course the EPROM's are more portable but use up the internal batteries far more quickly.

I checked out the LASER PC3 and it works and I got the cassette port to operate after I set the volume on the recorder at the proper level. It is more compact than the Z88 but I don't know how long a set of batteries will operate the unit.

**B**oth the Z88 and the PC3 have serial cables to a PC. Of course the PC end is a 25-pin connector since the technology was of the late 1980's. I have a cable made up by MR. FIXIT of Smart Computing to go between the Z88 and the LASER PC3. Mr. FIXIT is not on the staff of Smart Computing and is a helpful person much like the people with Timex/Sinclair computers. I will have to test out each cable. I have already tested the Z88 to PC cable but lost the upload. As I learn more about the workings of the PC maybe I can get it to work.

Seems like every time I try something that I end up

Chiricahua Apache, Left to right: Geronimo, Chihuahua, Nana, Loco, Josanja (Ulzana)



studying a half dozen books and get confused by all the information. But I slowly learn but get frustrated.0/0



# Seven-Bit ASCII Text Filter for the TS-2068 with Larken Disk I/O

by David Solly

Occasionally, after downloading a text file from an electronic bulletin board (BBS), or, converting HiSoft® PASCAL source code files into an LKDOS file, the file may fail to write to the screen, print, or load into M-Script<sup>1</sup>. This kind of problem is most often caused by rogue bytes which enter the text through line noise, the process of conversion, or perhaps from the operating system of another word processor, and interfere with the Sinclair operating system. The following Pascal program, once compiled, will quickly load an LKDOS file or an ASCII text file, remove from the file any rogue bytes encountered, compact the file, and provide the user with option of saving to disk by overwriting the corrupted file or saving under a separate name. The PASCAL source code listed below was treated in this manner.

## LKDOS Files from Sinclair BASIC listings

A Sinclair BASIC listing may also be converted into an LKDOS file, however, the resulting file may still fail to load into M-Script or print on a full-sized printer. This is because in addition to the possibility of having picked up a few rogue bytes during conversion the listing may contain Sinclair block graphics and user defined graphics. These graphic characters pose a special problem because they fall neither within the range of 7-bit ASCII nor conform to any 8-bit ASCII standard. It is necessary, therefore, to bring all Sinclair graphics into the range of 7-bit ASCII before applying the filtering process by using this additional piece of code

```
1200 CONST
1205 OFFSET = 79; (* Sinclair
      graphic to 7-bit ASCII *)
1400 IF (LETTER IN [128..164]) THEN
1410   POKE((START +
      COUNT), (LETTER - OFFSET));
```

The final result of this piece of coding is to change all user defined graphics from UDG 'A' through to 'U' into normal ASCII 'A' to 'U' and swop all the block graphics for a character between 'l' and '@'.

## Seven Bit ASCII Filter

*Program By David Solly*

```
100 (*$I,-*) (* COMPILE OPTION, P. 35 IN
MANUAL *)
110
120 PROGRAM ASCIIIFILTER;
122
123
125 (* STORAGE LOCATIONS FOR LKDOS
PARAMETERS *)
130 CONST
140   PROGNM = 23300;
150   PROGBGN = 23311;
160   PROGLN = 23313;
170
180   START = #A028; (* TEXT FILE LOAD
BUFFER *)
190   NEWSTART = #9C40; (* TEXT FILE COPY
BUFFER *)
```

```
200
210
220 VAR
230
240   ANS : CHAR;
250   PRINTABLE : SET OF CHAR;
260   NEWLN, BYTES : INTEGER;
270
280
290 PROCEDURE DOUT; (* INVOKES LKDOS
WRITE *)
300   BEGIN
310   INLINE (#F3, #CD, #62, #00, #21,
#04, #5B, #11,
320   #22, #20, #01, #09, #00, #7E, #FE,
#00,
330   #20, #02, #36, #20, #ED, #A0, #78,
#B1,
340   #20, #F3, #3E, #0B, #32, #02, #20,
#CD,
350   #CC, #00, #2A, #0F, #5B, #22, #33,
#20,
360   #2A, #11, #5B, #22, #31, #20, #CD,
#CF,
370   #00, #3A, #64, #00, #FB);
380 END;
390 400
410 PROCEDURE DIN; (* INVOKES LKDOS READ
*)
420   BEGIN
430   INLINE (#F3, #CD, #62, #00, #21,
#04, #5B, #11,
440   #22, #20, #01, #09, #00, #7E, #FE,
#00,
450   #20, #02, #36, #20, #ED, #A0, #78,
#B1,
460   #20, #F3, #3E, #0B, #32, #02, #20,
#CD,
470   #C6, #00, #2A, #0F, #5B, #22, #33,
#20,
480   #2A, #11, #5B, #22, #31, #20, #CD,
#C9,
490   #00, #3A, #64, #00, #FB);
500 END;
510
520
530 PROCEDURE SPOUT (C : CHAR); (*
PASSES CODES TO THE ROM *)
540 BEGIN
550   INLINE (#FD, #21, #3A, #5C, #DD,
#7E,
560   2, #D7)
570 END;
580 590
600 PROCEDURE AT (X, Y : INTEGER); (*
EMULATES BASIC 'AT' *)
610 BEGIN
620   SPOUT (CHR(22));
630   SPOUT (CHR(X));
640   SPOUT (CHR(Y));
650 END;
660
670 PROCEDURE CLS; (* EMULATES BASIC
COMMAND *)
680 BEGIN
```

```

690 PAGE;
700 SPOUT (CHR(22));
710 SPOUT (CHR(0));
720 SPOUT (CHR(0));
730 END;
740 750
760 PROCEDURE GETFILE;
770
780 VAR
790  FILENAME : ARRAY [1..9] OF CHAR;
800
810 BEGIN
820  WRITELN;
830  WRITELN ('FILE TO RETRIEVE FROM
DISK? ');
840  WRITELN;
850  READLN;
860  READ(FILENAME);
870  POKE(PROGNM,FILENAME);
880  WRITELN('HOW MANY BYTE IN FILE? ');
890  WRITELN;
900  READLN(BYTES);
910  POKE(PROGLN,BYTES);
920  POKE(PROGBGN,START);
930  DIN;
940
950 END; (* GETFILE *)
960 970
980 PROCEDURE STOREFILE(NEWLN : INTEGER);
990
1000 VAR
1010
1020  FILENAME : ARRAY [1..9] OF CHAR;
1030
1040 BEGIN
1050
1060  WRITELN;
1070  WRITELN;
1080  WRITELN('NEW NAME FOR FILE? ');
1090  WRITELN;
1100  READLN;
1110  READ(FILENAME);
1120  POKE(PROGNM,FILENAME);
1130  POKE(PROGBGN,NEWLN);
1140  POKE(PROGLN,NEWLN);
1150  DOUT;
1160
1170 END; (* STOREFILE *)
1180
1190 PROCEDURE FILTER(VAR NEWLN :
INTEGER);
1200
1210 VAR
1220
1230  LETTER, COUNTER, SCRATCHED :
INTEGER;
1240
1250 BEGIN
1260
1270  PRINTABLE := [' '..'];
1280  COUNTER := 0;
1290  NEWLN := 0;
1300  SCRATCHED := 0;
1310
1320  CLS; AT(11,3);
1330  WRITE('WORKING! KEEP YER SHIRT ON!');
1340
1350
1360 REPEAT
1370

```

```

1380  LETTER := PEEK((START +
COUNTER),INTEGER);
1390
1400  (* ENTER HERE ANY SPECIAL ROUTINE *)
1410  (* REQUIRED TO HANDLE SINCLAIR *)
1420  (* GRAPHICS OR 8 BIT CHARS *)
1425
1430  IF (CHR(LETTER) IN PRINTABLE) OR
1440  (LETTER IN [13, 27]) THEN
1450    POKE((NEWSTART + NEWLN),LETTER)
1460  ELSE
1470
1480    SCRATCHED := SCRATCHED +1;
1490
1500    COUNTER := COUNTER + 1;
1510    NEWLN := COUNTER - SCRATCHED;
1520
1530  UNTIL COUNTER = BYTES;
1540
1550
1560  CLS; AT(11,12);
1570  WRITE('FINISHED!');
1580  AT(15,0);
1590  WRITELN('NO. OF LETTERS REMOVED: ',
SCRATCHED);
1600  WRITELN;
1610  WRITELN('NO. OF LETTERS COPIED: ',
NEWLN);
1620  WRITELN;
1630  WRITELN('ORIGINAL LETTER COUNT: ',
BYTES);
1640  END; (* FILTER *)
1650 1660
1670 BEGIN (* MAIN PROGRAM *)
1680
1690 REPEAT
1700
1710  CLS; AT(5,0);
1720  WRITELN('SEVEN BIT ASCII FILTER');
1730  AT(7,0);
1740  WRITELN('THIS PROGRAM IS FOR
REMOVING');
1750  WRITELN('STRAY CHARACTERS FROM AN
ASCII');
1760  WRITELN('FILE WHICH MIGHT
INTERFERE');
1770  WRITELN('IN THE NORMAL OPERATION OF
A');
1780  WRITELN('WORD PROCESSOR. ');
1790
1800  GETFILE;
1810  FILTER(NEWLN);
1820  STOREFILE(NEWLN);
1830  WRITELN;
1840  WRITE('ANOTHER FILE? ');
1850  READLN;
1860  READ(ANS);
1870
1880  UNTIL (ANS IN ['N', 'n']);
1890
1900  WRITELN;
1910  WRITELN('TYPE RUN TO RESTART
PROGRAM');
1920
1930 END.

```

---

LKDOS is able to convert Sinclair BASIC listings and some other types of tokenized texts into straight ASCII text and save them to disk as a sequential file. For this article they will be called LKDOS files.

It's impossible to calculate it yourself, because any programs you right will mean you get the wrong answer. The only way I can think of to get this value is to load the game as normal, and stop the game with a Multiface as soon as the timer hits 000. Then put a breakpoint at #8C35 and return. Wait a few seconds, then reactivate the Multiface and have a look at the stack. The first value will be #8C35, the second will be the value of HL you want. You should find it's #4DBD. Carrying on the disassembly..

```
8C35 21 00 58 LD HL,#5800
8C38 11 01 58 LD DE,#5801
8C3B 01 FF 02 LD BC,#02FF
8C3E 36 00 LD (HL),#00
8C40 ED B0 LDIR
8C42 E1 POP HL
```

This clears the screen and restores the value of HL, which is used for the following decrypter.

```
8C43 11 60 8C LD DE,#8C60
8C46 0E 29 LD C,#29
8C48 7C LD A,H
8C49 65 LD H,L
8C4A 47 LD B,A
8C4B 09 ADD HL,BC
8C4C 1A LD A,(DE)
8C4D AD XOR L
8C4E 12 LD (DE),A
8C4F 6F LD L,A
8C50 13 INC DE
8C51 1A LD A,(DE)
8C52 AC XOR H
8C53 12 LD (DE),A
8C54 13 INC DE
8C55 CB 7A BIT 7,D
8C57 20 F0 JR NZ,#8C49
8C59 FB EI
8C5A 67 LD H,A
8C5B 11 70 71 LD DE,#7170
8C5E 19 ADD HL,DE
8C5F E9 JP (HL)
```

First of all, POKE #8C40, #8C41 and #8C42 with #21, #BD and #4D respectively (so you get the right value of HL), put a breakpoint at #8C5B (nearest place possible to the JP (HL) that we can place a breakpoint), and JP #8C40. On return, the value of HL is #38F5. Add this to #7170 (which is what happens in the next two commands) to get #AA65. This is the start address of the game. So put a JP to your POKEing routine (anywhere from #5B00 to #5BA0 is fine) at #8C5B, and finish your POKEs with a JP #AA65.

You will have to do a stack trace to find infinite lives in the actual game itself. There is a complete hack for this game by myself in YS #78, so why not disassemble it and have a look. It's slightly different to what we've done in that it intercepts the RET at the end of the loading system rather than mimic the first headerless loader, and puts a JP back to the hack at #83C3, but apart from that, it's more or less everything we've discussed above put together.

## PAUL OWEN'S PROTECTION SYSTEM

This has been used on a few Ocean games, but is in fact a standard headerless loader in disguise. The value of

A to use is always #98. Load in the BASIC loader and the first block of code, then stop it with a Multiface, and use a stack trace to find out the values of IX and DE for each block, and the JP to the game.

## SPEEDLOCK

Concluding the look at protection systems, I think it's only fitting that we end in quite possible the most famous protection system of all time.

Speedlock was first written by two guys called David Looker and David Aubery Jones around late 1983, although it wasn't commercially used until October 1984, on Daley Thompson's Decathlon, by which time it had reached it's third version. Since then, it has been used by many major software companies, especially Ocean. Its also gone many modifications, and can be split into three distinct generations.

I should state at this point that you need to have a Multiface to crack most of these Speedlocks, because they completely disrupt the operating system which will lock up any disassembler which relies on ROM routines. The Multiface relies on its own ROM, which isn't affected by the Speedlock code.

Type 1 - have one or two BASIC loaders, and load the main code with the infamous "clicking" leader tones (you know, instead of a steady "bleep", they go "blip, blip, blip, blip" a few times)

Type 2 - have one short BASIC loader, a long CODE block, lots of annoying beeps, then a similar loader to Type 1, minus the clicking leader tones, plus a countdown timer.

Type 3 - as for Type 2, except there is just one very long BASIC loader. The protection system crashes if a Multiface is left switched on. Mazemania on YS #77 covertape used a Type 3.

So, let's start at the very beginning (a very good place to start) with Type 1. In fact, there are about four different difficulty levels of Type 1 Speedlocks; the difficulty goes in chronological order (as you might expect).

The very first Type Ones were completely different to later ones, having the same initialization routine, but a completely standard decrypter. The only differences between this Speedlock and an ordinary decrypting loader were its initialization routine and its use of the IY register.

We came across index registers when we first met headerless loaders. There are, in fact, two index registers, IX and IY. In BASIC, the IX register is free for use in a machine code program run from a USR command, but the IY register must always contain the value #5C3A, which is the base address of the BASIC system variables which are wiped with a NEW command. If you return to BASIC with the value of IY anything other than #5C3A, the computer will crash, even if you use the "exit to BASIC" feature on a Multiface. The value of IY must also be #5C3A whenever a BASIC interrupt occurs. Both Devpac and 007 Disassembler run under the BASIC interrupts. They also use built in ROM routines, such as those to check the

keyboard and print text, this is preferable, otherwise they'd have to waste memory rewriting their own versions of the routines. Hence the value of IY must always equal #5C3A. The only safe way of using the IY register is to disable interrupts and write the whole program in RAM without using any built in ROM routines. And Speedlock fits this bill exactly, so it uses the IY register for most of its decrypter calculations.

Speedlock code also uses a lot of undocumented instructions. In theory, you cannot split the sixteen bit IY register into two eight bit registers. But the processor doesn't understand this, and you can split the IY register into two if you want. You simply put the code #FD on the front of any instructions using H or L. There are no standard names for the two halves of the IY register; but I will refer to them as IYH (Hi part of IY) and IYL (Lo part of IY).

Now let's hack a Speedlock game. To start with, I'm hacking Knight Lore, but the following games are also suitable: Beach Head, Daley Thompson's Decathlon, Gilligan's Gold and Underworld. Anything released after these will be explained later.

So first \*Hack the BASIC loader.

```
KNIGHT LINE 0 LEN 1037
0 BEEP 0.1,1:BEEP 0.1,2:BEEP
0.1,3:BEEP 0.1,4:BEEP 0.1,5:PAPER
0:BORDER 0:INK 0:BRIGHT 1:CLS:PRINT
BRIGHT 1;INK 0;AT 9,5;
"LOADING: KNIGHT LORE";AT
12,10;"PLEASE WAIT"
0 POKE (PEEK 23641+256*PEEK
23642),PEEK 23649:POKE (PEEK 23641+
256*PEEK 23642)+1,PEEK 23650
0 POKE (PEEK 23613+256*PEEK
23614),PEEK 23627:POKE (PEEK 23613+
256*PEEK 23614)+1,PEEK 23628
0 POKE 23662,PEEK 23618:POKE
23663,PEEK 23619:POKE 23664,PEEK 23621
23676 "REM CLOSE #ATTR...."
```

Wait a minute, there's absolutely no sign of a RANDOMIZEUSR command anywhere! There's just some BASIC which beeps a bit, sets the colours, and prints a message, a whole load of POKES, and then a load of garbage. Surely the computer will do everything, then report with an error message as soon as it reaches 23676? Well, that's not actually the case. Look at the third line 0 (the one which starts POKE [PEEK 23613+256\* etc.). This system variable is known as ERR SP. What happens is that when an error occurs (and it will do here), the computer jumps to the value in this register. This value is PEEK 23627+256\*PEEK 23628 PRINT this value, and there's the start of the machine code. You might get a different result to me, but I made the start address #60A8. Disassemble this address.

```
60A8 F3 DI
60A9 FD 25 DEC IYH
60AB FD 7C LD A,IYH
60AD FD AD XOR IYL
```

The DI is very important, because otherwise the I register can't be used. Given that IY starts off as being #5C3A, the value in A will end up being #5B XORed with #3A, which is #61.

```
60AF FD 26 F3 LD IYH,#F3
```

```
60B2 FD 2E A6 LD IYL,#A6
60B5 3B DEC SP
60B6 3B DEC SP
60B7 01 54 FE LD BC,#FE54
60BA FD E3 EX (SP),IY
```

First of all, this loads IY with the value #F3A6. It then decreases the stack pointer by two. By doing this, the stack pointer is now pointing to the start address of the machine code, which is #60A8. EX (SP),IY is a variation to the register exchange commands we've already come across. It basically swaps the value in the address pointed to by the stack pointer with the value in the IY register. So, after this instruction, IY will contain #60A8, and the value on the top of the stack will be #F3A6.

```
60BC 21 30 F2 LD HL,#F230
60BF FD 09 ADD IY,BC
60C1 01 AC 01 LD BC,#01AC
60C4 FD 5D LD E,IYL
60C6 FD 54 LD D,IYH
60C8 EB EX DE,HL
```

Here, HL is being loaded with #F230. The value in BC (#FE54) is added to the value in IY (#60A8), making the value in IY #5EFC. Then BC is loaded with #01AC, and the value in IY is transferred into DE. Then the values of DE and HL are swapped. So, by the end of the code we've looked at so far, HL will equal #5EFC, DE will equal #F230, BC will equal #01AC, and A will equal #61. These values are all used in the decrypter which follows.

```
60C9 AE XOR (HL)
60CA 12 LD (DE),A
60CB 7E LD (HL),A
60CC 23 INC HL
60CD 13 INC DE
60CE 0B DEC BC
60CF FD 6F LD IYL,A
60D1 78 LD A,B
60D2 B1 OR C
60D3 FD 7D LD A,IYL
60D5 20 F2 JR NZ,#60C9
60D7 C9 RET
```

This is a straightforward decrypter, except the value for A (which is needed throughout the decrypter) is temporarily stored in part of the IY register. The RET is to #F3A6 (the top value on the stack).

To crack this, we can set up the register values manually, CALL the decrypter, and then hack the main loader ourselves. Type out this program:

```
5B00 F3 DI
5B01 21 FC 5E LD HL,#5EFC
5B04 11 30 F2 LD DE,#F230
5B07 01 AC 01 LD BC,#01AC
5B0A 3E 61 LD A,#61
5B0C CD C9 60 CALL #60C9
5B0F FD 21 3A 5C LD IY,#5C3A
5B13
```

Notice that we've disabled interrupts to avoid crashing, and we need to restore the value of IY to #5C3A afterwards, so your disassembler won't crash. RUN the program, and have a look at the code at #F3A6. You'll find it's just a straightforward headerless loader with absolutely no frills, and you should be able to hack it no problem. As for the final hack, load the BASIC into address #5CCB, run the decryption routine above, patch the JP in the main turboloader to your POKES, and start running.



All other Speedlock Type 1s have the same sort of decrypter. The code for the decrypter is very complicated, with the result that I have been unable to reproduce it here. Luckily, you don't have to touch the code; you can write your own decrypter as long as you have a Multiface.

I'll be doing Tapper as an example, but any other Speedlock follows this procedure almost exactly. \*Hack your game and note down the length of the code (you'll need it later). I made it 1453, which is #05AD hex. Now look at address #5EFD. The byte at #5EFD is always decrypted to give the byte #42, and the byte at #5EFD is always decrypted to give the byte #55. The decrypter works by XORing the encrypted byte with a number taken from the R register. By inspecting the code before and after running, you'll see the XORing number starts off as #CB, and increases by #0A each time. If the result is more than #FF, the result has #80 subtracted from it. We can incorporate this into our decrypter. The start of the code is #5EFD, and the length is (PEEK 23627+256\*PEEK 23628)-#5EFD, which is #01ED in the case of Tapper. The following code will simulate the decrypter.

```
LD HL,  
LD BC,  
LD D,  
*** LD A, (HL)  
XOR D  
LD (HL), A  
LD A, D  
ADD #0A  
SET 7, A  
LD D, A  
INC HL  
DEC BC  
JR NZ, ***
```

Once you've done that decrypter, you've got to do the whole lot again, starting at #5EFD. The byte there will be decrypted to either #3E or #ED - you'll have to guess which decrypting value to use. For Tapper, the start is #5F2B, the length is #1BF, and the second decrypter value is #AB.

When you've done that, you'll either get the complete loading system or another decrypter. If you've got the loading system, then reload the BASIC loader, and do a stack trace to find out where it should go. You should have no problems with the loader.

If you've got another decrypter, go along five bytes and find a LD DE,(XXXX). Add #2E to this value, and that's where you move it to. The length is the same as that for the second decrypter. The decrypter itself can be cracked by changing a JP Z in the code about forty bytes later (the value of this is the start of the turbo-loader), but the decrypter itself uses a byte which is worked out by adding all the memory together in the loading system. Since we've got an exact copy of this system elsewhere in memory, just change the value of XXXX in the aforementioned LD DE,(XXXX), and then JP to the start of the decrypter

If the first decrypting value you used was #CB, then you can just change the JP in the turbo-loader to your POKEs.

If the value was #CD, then you'll need to know about the Standard Speedlock patch. Somewhere in the loading

system there will be the two bytes ED 53 [LD DE,(XXXX)]. Change the XXXX to the address of your POKEs (#5BA0 is normally safe), and end your POKEs with a JP to the value you overwrote. You'll have to use this patch for the later Speedlocks as well.

There was a Speedlock Type 1 MultiPOKE in YS#79. RUN the program, press BREAK and disassemble address #5B00 to find out what to do in your own hacks. Type 2 Speedlocks feature a very easy BASIC loader, and one big block of code, which has six short decrypters and a complex moving routine. The decrypters are all easy peasy, just move them to somewhere else in memory (such as #5B00), bung a RET on the end, and CALL the decrypter from there (but watch out for the third decrypter, which checks for a Multiface and crashes if it finds it. The moving routine fiddles about with the loader. Search for 31, which means LD SP,XXXX. Hopefully, you'll find a LD SP,#0000, with perhaps a DI right before it. Write down the address and run the moving routine (you may have to restart the tape, because some of the moving routines insist on a signal coming into the tape recorder). Use a stack trace to find out where the code has gone to. Now you can move all the code from the moving routine to the end of the machine code block to where it should be, given that you know where the LD SP,#0000 goes to. Once moved, patch the loader in the same way as the first Speedlock.

Type 3s have just one long BASIC loader, with about 144 decrypters, but that's nothing to worry about. \*Hack the BASIC loader, and have a look at the first bit of code which moves the rest of the code into the right place (you can then use a headerless loader to load this into the right place in memory). The tricky bit is changing a byte in memory so a CALL to the loading system at the very top of the code is changed to a CALL somewhere else once all the decrypters have been run. The only way you can do this is to change the address of the hi byte of this CALL to something else, and RUN the huge load of the decrypters. The computer will crash if you have a Multiface attached, but only after everything's been decrypted, so then look and see what the CALL's been changed to. If it's suitable, remember the patch, position your hack around this, change the CALL to what it should be, and put in the usual Speedlock patch. Look at the start address in the turbo-loader. This address will be overwritten by a decrypter once loading finishes. This decrypter is nothing special, so just crack it as usual, and watch out for the game moving around.

On North's Pokerama Tapes usually have a Speedlock Type 3 crack on it - load up the Pokerama, choose your POKE, then do a stack trace to find it and have a look at it.

## Part 7 - Epilogue

Well folks, I'm sorry to have to break this to you, but I've just about told everything you should ever need to know to crack every protection system under the sun. So I'll just say some final words and credits, and then sign off, okay?

The idea of this book, its production and its writing were done entirely by Richard Swann, from February to June 1992.

Some suggestions and tips came from two people to whom this book is dedicated, Matt Corby from "down-the-road", and Niall "Mr Incredibly Technical" Daley. Thanks, guys!

Thanks to YS for putting disassemblers and the like on recent covertapes; it saved me the trouble of writing one!

Thanks to Jon North for some of the info on hacking he gave to me on disk recently - much appreciated, mate.

Thanks for YOU for buying this. There aren't many Speccy hackers around right now, so we need to make the numbers up. Good luck!

What's that? "I don't understand this bit at all!" I hear you say. Well, send me any queries that you may have about this book, stating exactly what the problem is, along with an SAE (very important that), and I'll do my best to reply to them. DON'T write to me asking me to hack a whole list of games for you - I just haven't got the

time. However, I've got a big book of Multiface POKEs which you can obtain for £1.50 and an A4 SAE if you want it, so that might come in handy.

Thought for the year: Seven years ago, Spectrum and Commodore owners were at each other's throats. Spectrum owners would vow never to have anything to do with Commodore. Then why do I hear of so many Spectrum owners that have upgraded to a Commodore Amiga? Personally, I can't stand the Amiga's operating system - it's terrible!

Well, that seems to be about it, so I'll just leave you know in the hands of a glossary of terms. Happy hacking!

Richard Swann - June 1992

## THE END

[C] 1992 NSA Publications. No part of this book may be copied, otherwise I'll send the Mafia round. Okay?

# Memory Improves With Age by the late William Pedersen

Crazy? Perhaps so, but survival of our favorite antique TS2068 computer depends on it to a degree. Two advances are primarily responsible for this. Both extend the amount and speed of available memory, over and beyond that in the DOCK bank.

Disk Drives improve speed, accessibility and convenience over tape recording.

RAMDISK has broken all speed records and has expansion possibilities which are impressive. Now all we need to do is find some way to bank switch additional memory. It has been said it can't be done because Timex expertise has been dispersed - *dinosaur chips*!

While it is true that the system Timex intended (before killing it) is highly complex it is NOT the only system which works. The one described in Figures 1 and 2 is just about the minimum bank switching system. There are a lot of features like handling interrupts and auto-configuring that are beyond it, but it can address 16 megabytes of memory and works with the unmodified TS-2068.

## Memory Bank (fig. 2)

There is nothing particularly special about this memory bank. It decodes a 24 bit address and has a dip-switch to locate 64K of contiguous memory space at any 64K boundary.

If any of its memory is active, it drives the BE signal low to disable LOCAL banks (HOME, EXROM and DOCK). With this system it is impossible for two expansion banks to be on the

bus at the same instant. (See Appendix A for greater detail.)

## System Configuration

There must be some way to MAP all or part of memory.

The simplest way to do this is to write programs which assume continuous memory up to a variable limit, and warn you when that limit is exceeded. For this, each expansion bank should have dip-switches set from 1 to MAXBANK in sequence.

There are better ways which allow unused available banks to be switched around where needed; saving \$\$\$\$\$. This needs some programming overhead, but you have *room* for it.

## Auto Configuration

This is where the computer is programmed to go out to explore the neighborhood. Whatever it finds is recorded into a system configuration table (SYSCON) for later reference. *This is not easy!*

A **dumb** device cannot be found because it cannot answer a roll-call. It is deaf to attention-getting methods. It can be made visible by attaching a baby-sitter chip which can respond and identify itself and the attached device.

A **smart** device listens, answers, and often calls for attention. Still, it must also accept an order to shut up so it won't interrupt. The IEEE488 system is a system for standardizing command language between widely different computers and devices, but it is still not smart enough to keep quiet when another device with

the same "name" is on the system bus.

Duplicate names can be resolved if they are at different addressable locations. In this way one of the names can be changed to avoid future confusion and wasted time. It only needs to be done initially, or when some outside event has created another duplication (Like turning on a disk drive).

A **daisy chain** (USB) is one way to resolve the unique address problem.

Another common method is to give backplane slots an address on a temporary basis. (perhaps expandable using a **daisy chain**)

Both methods assume something about the design of the connecting **network**. It must guarantee reaching only one device at a time.

### **NetWorking**

Almost by definition, a network is where Murphy lives. Frequently the unexpected happens. No more than three points will be made here.

1. Statistical methods must be used to find time slots when bus confusion is absent by chance.

2. If this works, there is no need for physically unique addresses, though default names still help.

3. Any device connected to a network can help by introducing a 3 random delays to that natural to the system.

### **Bank Switching**

A bank switching controller (BSC) is essentially a network switchboard. The TS2068 operating system expects eight equal 8K wide channels assigned arbitrarily to **chunks**.

The Z80, like most CPUs, has channels to internal registers; and internally swaps between register sets. It is a bank switching controller itself.

Machine code includes extended addressing provided by additional fetched bytes. Prefix bytes allow instruction sets to be bank switched.

The 8088 CPU used in the IBM PC and clones has four dedicated internal bank switching registers whereas the Z80 has none. This idealistic approach for the 8088 worked like a charm until it ran into deep water. It outgrew its island and couldn't build a boat. Externally, bank switching registers do not have these limitations.

The BSC in Figure 1 is similar to the one inside the 8088, but there the resemblance ends. It has eight channels instead of four, and can address sixteen times as much space.

There are better BSCs than the one in Figure 1, but this one does a powerful job. It also requires no alteration of your precious TS2068, a good compromise.

### **Bank Switching Operation**

The most significant three bits of Z80 address space are used to select one of eight previously established address extension bytes in a current chunk owner table (CCOT). The remaining bits address locations within each 8K channel. When power is first turned on, and at other times when the TS2068 needs exclusive control, CCOT must be turned OFF. Turning it ON could be a problem.

Fortunately we can first assign all eight channels to HOME. The extended address is 255 for which nothing usually responds. Then nothing happens when CCOT is turned ON. (The "usually" reference is explained later.)

With CCOT turned OFF extended address bits float, unless something is connected to force them high. In effect, the OFF condition also gives 255 for the extended address. That is exactly why HOME bank was assigned that bank number.

### **Bank Switching Control**

Except during power-up there is no safe place in physical memory to put bank switching code. Only the fetched instruction in the Z80 is immune. That instruction must be able to find the BSC regardless of memory assignments. It must use I/O (another example of Z80 bank switching). Because GO TO and CALL are not I/O instructions, and the machine stack is unsafe anyway, each bank chunk using these needs to support them with MACROS having the same effect. The same applies to the RETURN instruction.

The most elegant method avoids using these by continuing the code in the shadowing bank and leaving the chunk with the current machine stack untouched.

This flexibility is what makes multitasking and multi-user time sharing systems remarkably easy to create. Each user can have his own private partition under control of a SUPERVISOR. Security from program interference remains a

problem with the Z80, not like other CPUs which provide privileged instructions. Certain programming conventions will have to be followed to compensate for this.

There is great pressure to reserve one chunk to one bank to hold the system variables (SYSVARs), machine stack and bank switching routines. This is the technique used in the relics of machine code remaining in the unmodified TS2068. If repeated in each user's partition, this is a good convention to use. It should be remembered that it is still a convention, and need not be followed.

The relics use memory mapped bank switching ports. Unless these port addresses are forbidden to be used in all banks, it doesn't work. The BUG consists of stacking the port byte from one bank and restoring it in another, thereby destroying it in the new bank. There is a way to switch stacks to resolve this problem, but is too complex for serious consideration.

Global SYSVARs can be stored in I/O space, so this is not a real difficulty. The same applies to a bank switching stack run by the MACROS already mentioned. An alternative for the bank switching stack is to dedicate a fixed bank and chunk for it, though this has the problem of not remaining safe from accident.

Some of this does not apply to the BSC in Figure 1. It has been included to show what can be done with a more advanced version.

Control is simple. This BSC is write-only. It has eight ports of which only four are presently used. The port assignments avoid those to which the ZX and TS2040 printers respond and all known physical interface ports like the MODEM, AERCO, LarKen, A&J, and TASMAN.

PORT	DATA	FUNCTION
132	x	Turn CCOT OFF
133	x	Turn CCOT ON
134	Bank#	Owner to be posted to CCOT
135	"HS"	CHUNK mask, active LOW.

Bank # is identical to the extended address byte. OUT 134, Bank # writes the bank number into a register which maintains it as input to CCOT. "HS" is the "Horizontal Select" described by TIMEX. Looking at the structure of CCOT, it makes sense.

OUT 135, HS latches the stored Bank # into CCOT registers for which HS bits are active LOW.

Some of the more significant bits can be ignored in smaller systems, which explains why EXROM is Bank # 254 and DOCK is Bank #0. A system using only the lower nibble can still address 1 megabyte of memory.

Control can be safer from accident by using one of the unused ports to act as "Simon Says". This was the real purpose of the **Timex Reset Nibble Sequence** bank switching instruction, though never explained.

### Exceptional Conditions

Bank numbers from 1 through 253 are gravy, but what about HOME (255), EXROM (254), and DOCK (0)? Can they be implemented as real external banks? The answer is definitely yes, with some limitations.

HOME ROM can be replaced with EPROM at the drop of a hat. An almost trivial case is replacing it with SPECTRUM ROM. What is NOT trivial is the resulting SPECTRUM having bank switching capabilities! Repairs to the TS2068 ROM code can be made in EPROM with impunity. Unlike other banks, bank 255 is active immediately at power-up. Because SPECTRUM is a subset of TS2068, it remains a toy with which to play. The real power lies in installing upward compatible extensions to BASIC and to restore TIMEX disabled functions like OPEN, CLOSE, RESET and CATalogue.

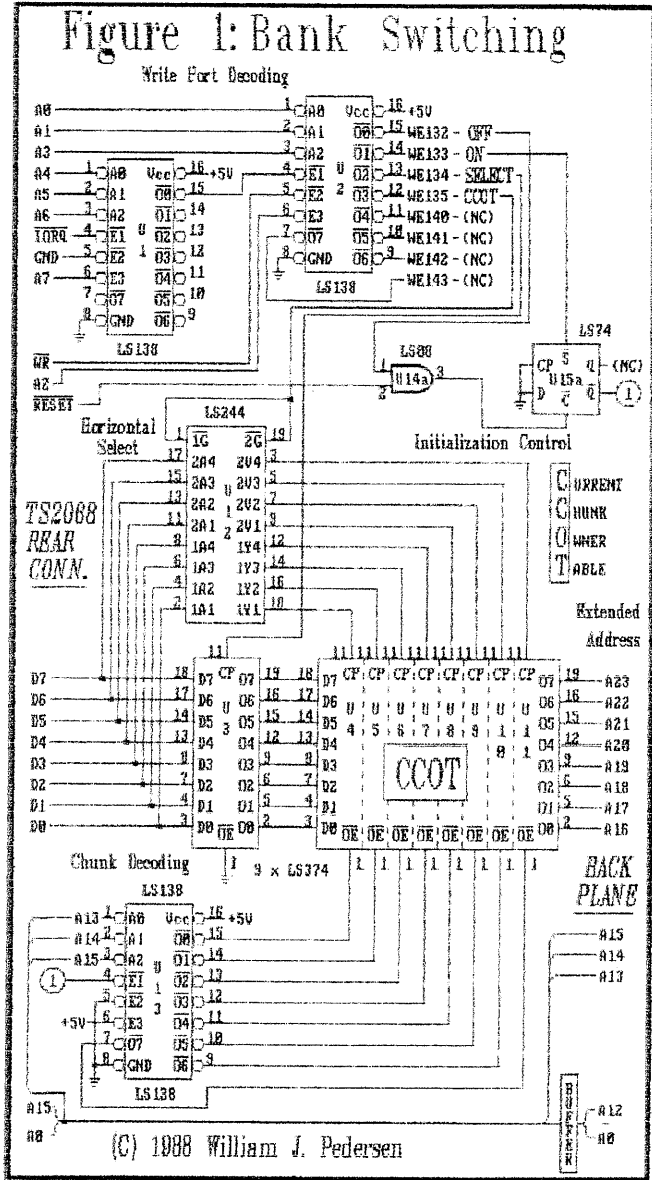
EXROM cannot be directly replaced without removing it from the case, and even then, it is not available during power-up. The problem with internal EXROM is that it is incompletely addressed so an image of it appears in all chunks, not just chunk 0.

The addressing problem can be solved by moving the chip to a special card which plugs into the cartridge slot. This card doesn't change anything except to provide full addressing to suppress the false images. Of course, if EXROM is replaced with EPROM you can make repairs to code. In any case, you now have chunks 1 through 7 of Bank # 254 (enabled by EXROM signal) which you are free to use. On the expansion bus Bank # 254 will have priority over this, but it is not active during initialization.

DOCK is usually enabled by the ROSCS



signal available at the cartridge slot. If you relocate it to the expansion bus as Bank # 0, it has priority when ON, or can act the same because the ROMCS signal is available on the expansion bus.



If you have no back-plane, the EXROM relocation card is the perfect place to include replacement HOME ROM using a technique described in another article for recovering the ROMCS signal logically--as it is not available at the cartridge slot.

For the same reason, this card should carry a replacement card edge connector 60 you can still use your cartridges.

Should you desire, once the system has been initialized, and you DO have an expansion bus, even these replacements can be replaced by the

external higher priority back-plane cards. When this is done, EXROM and DOCK are no longer mutually exclusive.

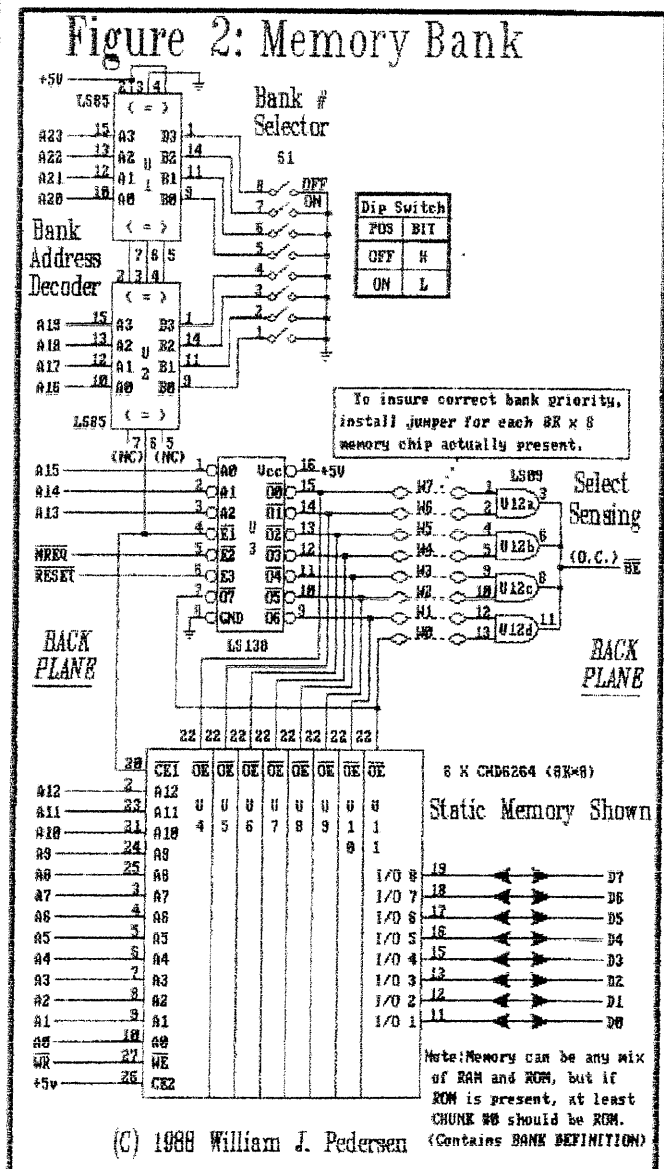
### Summary

While the Bank Switching System described in Figures 1 and 2 has limits on what it can do, it should start a revolution. It is simple enough to understand, uses easily obtainable parts, and any reasonably competent hardware hacker can build it. It doesn't put your TS2068 at risk, and even if EXROM is relocated, there are no wiring changes internally so you can always put it back.

A lot of mental sweat went into creating this version. The main reason for it is to give the user confidence that it can indeed be done as advertised. Perhaps later someone will actually believe that a full self-configuring system actually exists (which it does).

It nevertheless is one **giant step** for the TS2068. Once Taken --- Watch Out!

(C) 1987 William J. Pedersen WIDJUP



# Unclassified Ads

Place your ads here, it is free!

Mail/E-mail to: A. KAHALE 3343 S FLAT ROCK CT SIERRA VISTA AZ 85650-6874

Please inform and/or update the Editor of any changes in your ad/s

## Home Electronics Service

We have been a part of the Sinclair scene since 1982, repairing ZX Spectrums for Sinclair Research in England.

We provide Sales, Service, and Software for the

**QL, Spectrum, ZX-81 and Z88.**

[www.members.tripod.com/hes\\_computing/hes1.html](http://www.members.tripod.com/hes_computing/hes1.html)

E-Mail 74601.1535@compuserve.com

Hours of Operation is Monday - Friday 1300 hrs. to 2100 hrs. central time zone.

Phone 210 661-4376

**John R. Rish**

Home Electronics Service

5222 Kazen Dr.

San Antonio TX 78219 USA

Bill Mathis

## Pro Digital Electronics

323 SE 28th Ave.

Portland, Oregon 97214

503-232-3200

## PRO ACT CONSULTING, INC.

2660 N. Houghton Rd.

Tucson, AZ 85749

520-749-5395, fax 520-749-3626

email <proactmd@aol.com>

## The John Oliger Co.

11601 Widbey Dr.

Cumberland IN 46229

The John Oliger Floppy Disk System

**FOR THE TS-2068**

EXPANSION BOARD

2068 User Cartridge

DISK BOARDS "A" & "B"

2068 Parallel Printer Port

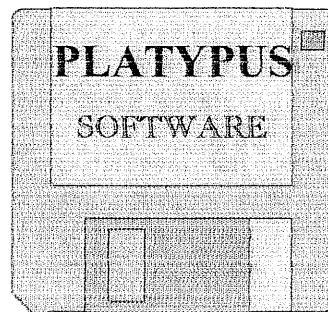
2068 EPROM Programmer

2068/SPECTRUM Joystick Port

DFh Mapped Universal I/O Port board

User Manual only : \$5.00 (Read before you buy)

[joliger@mindspring.com](mailto:joliger@mindspring.com)



QLAMBer \$20

QLuMSi \$20

SeekQL \$10

Upgrades \$5

*Al Feng*

914 Rio Vista Cir SW

Albuquerque NM 87105

(505) 843-8414

Make David Leech an Offer

ZX-81/TS-1000 TS-2068 Hardware Kits

Real Time Clock I/O Controller RS-232

Centronics I/F 16K & 64K RAM 300 BAUD

Modem A-D Converter(assembled)

**BYTE-BACK INC**

536 Long Ter

Leesville SC 29070

ZX-TEXT - Word Processor

ZX-CALC - Spread Sheet

**Business Software**

Cycle Accounting Financial Report Generator

ZX-CALENDAR - Time Management

ZX-81 TS-1000 TS-1500 TS-2068

Albert F. Rodriguez

**A.F.R. Software**®

1605 Pennsylvania Ave 204

Miami Beach FL 33139

305 531-6464

## WOOD & WIND

ARCHIVE Based QL Software

QLerk - A complete financial program for the QL

QLerk software (v3.21) with tutorial \$29

QLerk manual \$29

QLerk software & manual \$50

DBEasy - A menu based database system

DBEasy software (v1.6) \$24

DBEasy upgrade from V1.5 \$7

DBProgs - A toolkit of ARCHIVE procedures

DBProgs software (v1.8) \$18

DBProgs upgrade from V1.7 \$7

DBTutor - A general purpose learning program  
DBTutor software(v1.5) \$12  
PC DBEasy - Just like QL DBEasy  
PC DBEasy software (v1.3) \$12

Bill Cable  
**Wood & Wind Computing**  
RR3 BOX 92  
Cornish NH 03745 USA  
Phone (603) 675-2218

## QBox USA

810 254-9878

24 hours a day  
300 to 28,800 bps

Supporting all Sinclair and Timex users Fidonet Echomail arefor Sinclair computers. Lots of new files for you to download such as TS-2068 emulator for those who use a PC give us a call and let us know what you want to see Message and file areas

QL International, Quanta, QL Hacker's Journal, Spectrum/2068, ZX-81/TS-1000, Z88, NetMail, emulators, pointer, FDFormat for QXL/QDOS, etc.

**Sysop John J Impellizzeri**  
**Co-Sysop Don Waltermann**  
**Utica, Michigan, USA**

## Domino Cubes

### Z88

Hardware & Software

352 7<sup>th</sup> Ave. 15<sup>th</sup> Flr.

New York, NY 10001

Phone 212 631-7563

Fax 212 947-5069 Voice mail pager 917 490-8407

Domino.cubes@excelsior.net

## LISTing Newsletter

The Long Island Sinclair/Timex Users Group

Robert Malloy, Treasurer

412 Pacific St.

Massapequa Park, NY 11762

## NESQLUG

### NEWS

New England Sinclair QL Users Group

Ed Kingsley, Editor

16 Highland Avenue

Saugus MA 01906

(781) 233-3671 EdK4@aol.com

## QL Hacker's Journal

Supporting All QL Programmers

Timothy Swenson, *Editor*

2455 Medallion Dr.

Union City, CA 94587-1914

swensontc@geocities.com

<http://www.geocities.com/SilconValley/Pines/5865/>

Alec Carswell

## Motivation

16 Montgomery Ave

Beith Ayrshire KA15 1EL

United Kingdom

## The Rantop

*Newsletter*

The Greater Cleveland T-S User Group

Thomas Simon Editor

615 School Ave

Cuyahoga Falls OH 44221

E-Mail CIS 73177,333

Jon Kaczor Production

4568 Williamson Ave

Brooklyn OH 44144

75363.1127@compuserve.com

## ZX-TEAM MAGAZIN

Peter Liebert-Adelt

LUETZOW STR 3

D-38102 BRAUNSCHWEIG

GERMANY

Email: p.liebert@t-online.de

[http://home.t-online.de/home/p\\_liebert/zx-team.htm](http://home.t-online.de/home/p_liebert/zx-team.htm)

Amateur Radio: DK4BF@DB0FC.#NDS.DEU.EU

## The ZX Spectrum 48/128 Emulator for IBM & Compatibles: Z80 Version

Turn your PC into a real ZX Spectrum 48/128

⇒ Full Spectrum emulation, border, flash, beeper, Interface 1, Microdrive in cartridge file, RS232 input and output redirection to file, COM or LPT, joystick support, 128K sound through Soundblaster or internal speaker, built-in monitor,

⇒ Able to load ANY, even protected or speed-saved program from tape, to save to tape, to redirect tape loads and saves to disk for easy file access,

⇒ 2500 line English documentation.

⇒ Runs okay under DOS, Windows and DesqView,

⇒ Full source code of emulator and utilities included!

Runs on any 640K PC; too slow for practical use on PC/XT's but fast enough on AT's, uses VGA/EGA/CGA or Hercules.

This program costs US \$20. You will receive a 3.5" DD disk (5.25" disks on request), and you'll be kept informed about up-

dates. Please send bank notes (bills), name and address to:

**Gerton Lunter**

PO Box 2535  
NL-9704 CM Groningen  
Netherland

Check, please add US \$15 extra and allow 4 weeks for delivery.

**Jochen Merz Software**

The Fully Functional ROMDisk

**SMSQ/E for the QXL**

**SMSQ/E for the Super GoldCard**  
QL Games & Upgrades QL Applications

ProWesS + Applications

Jochen Merz Software

Im stillen Winkel 12

47169 Duisburg, Germany

☎ 0203-502011 Fax 0203-502012

Credit Cards accepted

<http://www.j-m-s.com/smsq/>

e-mail [smsq@j-m-s.com](mailto:smsq@j-m-s.com)

**QL Today**

QL Today is published by Jochen Merz Software. Jochen Merz has been supplying software for the QL for several years and has built up a good reputation for quality and fair trading. The representative in Britain is Miracle Systems Ltd. who take subscriptions and do the distribution.

**English Office**

Miracle Systems Ltd.

20 Mow Barton

Yates, Bristol

United Kingdom BS17 5NF

Tel. +44 1454 883602 Fax. +44 1454 883602

**Editor**

Dilwyn Jones

41 Bro Emrys

Tal-Y-Bont, Bangor, Gwynedd

United Kingdom LL57 3YT

Tel. +44 1248 354023 Fax. +44 1248 354023

**Items for the Timex\Sinclair Computer**

**Timeworks Programming kit #1** For T/S 1000 & ZX81 \$4.95

**Mindware Gulp Game** Timex 1000 & Sinclair ZX81 \$4.95

**Timex Horace & The Spiders** for the 2068. \$5.95

**Chess** (16K RAM) qty 5 price \$2.95 ea

**The Challenger I** qty 17 price \$2.95 ea

**Mindware MW100** plain paper printer for the TS1000 or

ZX81 computer made in the US. \$45.00

All items are new. Please add \$3.95 shipping to all orders.

MC, VISA, American Express. phone 717-748-1747

**Keith Electronics**

224 North Grove St

Lock Haven, PA. 17745

**ZX-81 Kit**

\$30 Plus \$10 for S/H

**TS-2040 Printers 16K RAM**

We Do Not Ship Outside The US & Canada

Checks, Money Orders and Visa or Mastercard

Call 212-675-8414 FAX 212-675-8980

Zebra Systems, Inc.

122 W26th St. Suite. 904

New York, NY 10001

Shipping is by UPS ground.

Basics of Timex Sinclair 1500/1000 BASIC

BASIC Basics for the Timex/Sinclair 1500/1000

The Ins and Outs of the Timex TS-1000 & ZX-81

Computer Interfacing Technique in Science TS-1500/1000

ZX-81 BASIC Programming (in Spanish language)

**Sinclair Resources**

John McMichael (Developer - Graphics)

1710 Palmer Dr

Laramie WY 82070

Keith Watson (AERCO & Z80 Emulator)

41634 Amberly Dr.

Mt. Clemens, MI 48038

Rod Gowen (RMG)

14784 S Quail Grove Cir

Oregon City OR 97045-8843

Send them a LSASE and ask for information about their current products and/or services.

**Surplus T/S Inventory**

JOHN J SHEPARD III

281 130<sup>th</sup> ST

OGDEN IA 50212

< [jshepard@netins.net](mailto:jshepard@netins.net) >

Mostly QL & TS-2068

JACK BOATWRIGHT

67325 FRYREAR RD

BEND OR 97701

< [jboatno4@outlawnet.com](mailto:jboatno4@outlawnet.com) >

Mostly ZX-81/TS-1000 & TS-2068

**QL TS-2068 ZX-81**

Books Magazines

**Softwar**